PRACTICA HIBERNATE

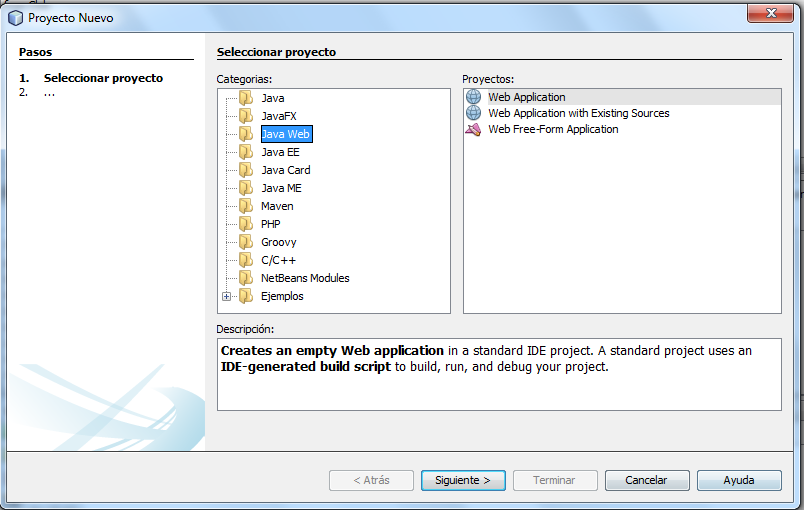
Introducción

Hibernate es una herramienta de Mapeo objeto-relacional (ORM) para la plataforma Java (y disponible también para .Net con el nombre de NHibernate) que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans de las entidades que permiten establecer estas relaciones.

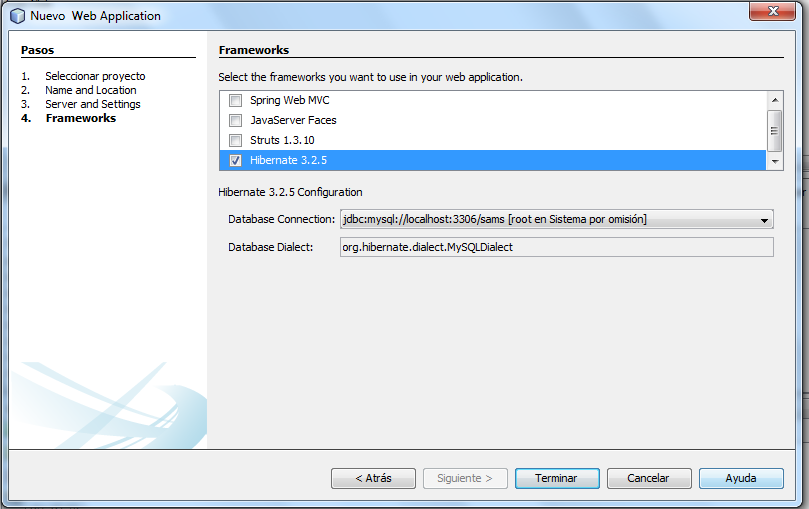
Hibernate es software libre, distribuido bajo los términos de la licencia GNU LGPL.

Como todas las herramientas de su tipo, Hibernate busca solucionar el problema de la diferencia entre los dos modelos de datos coexistentes en una aplicación: el usado en la memoria de la computadora (orientación a objetos) y el usado en las bases de datos (modelo relacional). Para lograr esto permite al desarrollador detallar cómo es su modelo de datos, qué relaciones existen y qué forma tienen. Con esta información Hibernate le permite a la aplicación manipular los datos de la base operando sobre objetos, con todas las características de la POO. Hibernate convertirá los datos entre los tipos utilizados por Java y los definidos por SQL. Hibernate genera las sentencias SQL y libera al desarrollador del manejo manual de los datos que resultan de la ejecución de dichas sentencias, manteniendo la portabilidad entre todos los motores de bases de datos con un ligero incremento en el tiempo de ejecución.

Creando el proyecto (Windows 7)

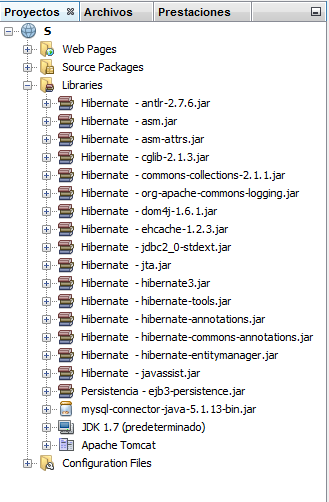


* Elegimos un servidor como Tomcat o Glassfish, Siguiente y marcamos la opción Hibernate y seleccionamos la base de datos jdbc:mysql://localhost:3306/sams en nuestro caso, de la lista que, supongo, ya habías configurado anteriormente como indiqué en los pasos previos de este tutorial.



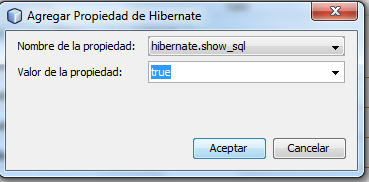
* Fíjate que en: **S** > Libraries

Se agregaron automáticamente las librerías necesarias para utilizar Hibernate y conectar a la base de datos.

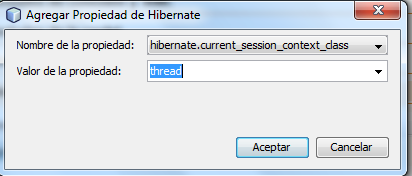


Configurando el proyecto

* Abre el archivo hibernate.cfg.xml que está en la ruta: S>Source packages <paquete predeterminado> > hibérnate.cfg.xml
* Una vez abierto en NetBeans IDE, en el modo Diseño, en la parte Propiedades opcionales, Propiedades de configuración, pon Agregar... y añade la propiedad:

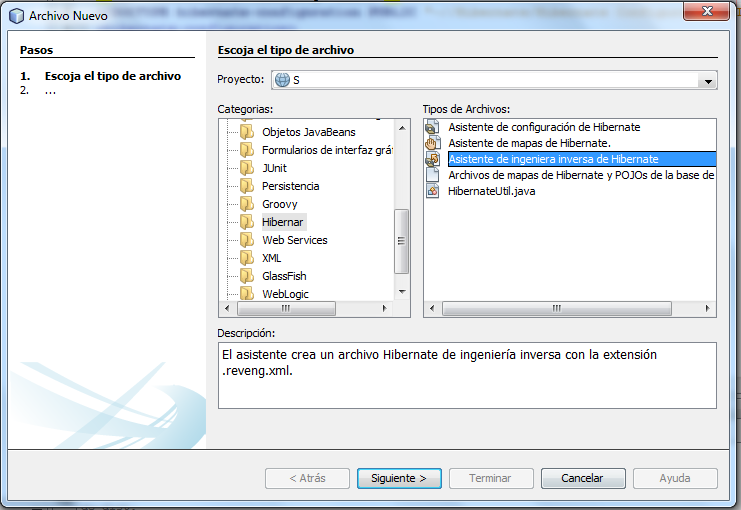


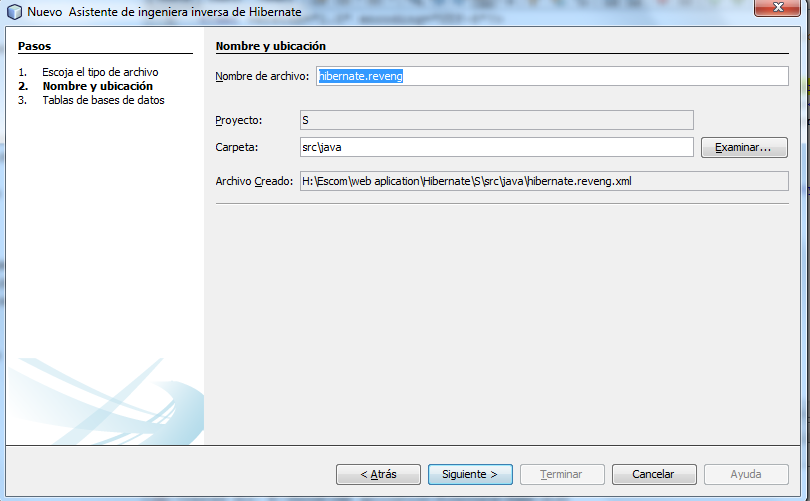
* Dentro de las mismas Propiedades opcionales del modo Diseño de NetBeans IDE, más abajo, en Propiedades varias, Agrega la propiedad:



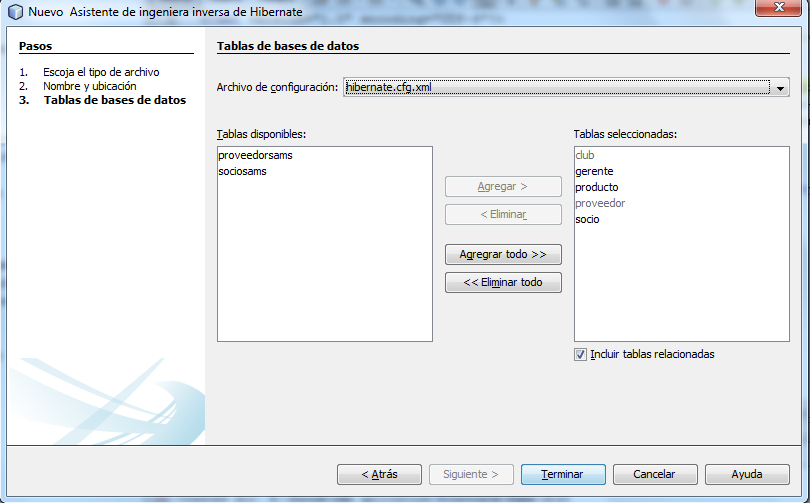
Asistente de Ingeniería Inversa de hibernate.

* Agreguemos un nuevo archivo, y seleccionemos Hibernar > Asistente de ingeniería inversa de Hibernate.





* En el siguiente paso dejamos los valores por defecto (asegurándonos que pertenezca al proyecto a **S**). damos clic a Siguiente



* Se cargarán automáticamente los nombres de las tablas de la base de datos S (elegida cuando creamos el proyecto), y seleccionaremos las tablas:
  + club
  + gerente
  + producto
  + proveedor
  + socio
* Si marcas la casilla Incluir tablas relacionadas algunas otras tablas se incluirán automáticamente. Eso está bien.
* Al presionar Terminar se va a crear un archivo hibernate.reveng.xml parecido a este (No hay que editarlo):

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE hibernate-reverse-engineering PUBLIC "-//Hibernate/Hibernate Reverse Engineering DTD 3.0//EN" "http://hibernate.sourceforge.net/hibernate-reverse-engineering-3.0.dtd">

<hibernate-reverse-engineering>

<schema-selection match-catalog="sams"/>

<table-filter match-name="producto"/>

<table-filter match-name="club"/>

<table-filter match-name="proveedor"/>

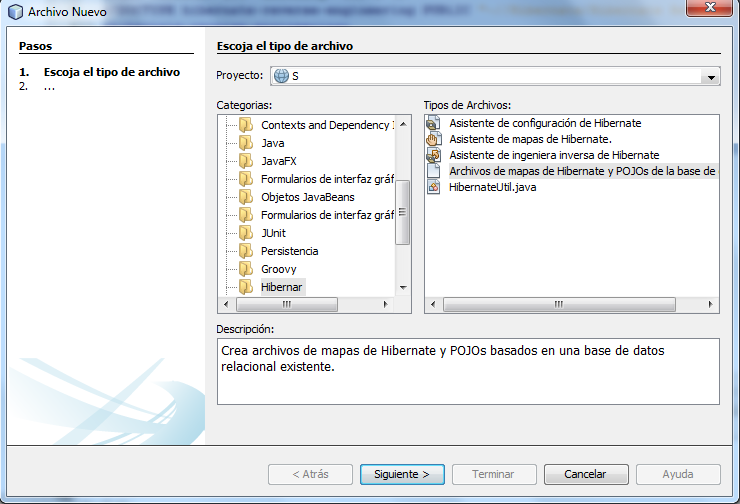
<table-filter match-name="socio"/>

<table-filter match-name="gerente"/>

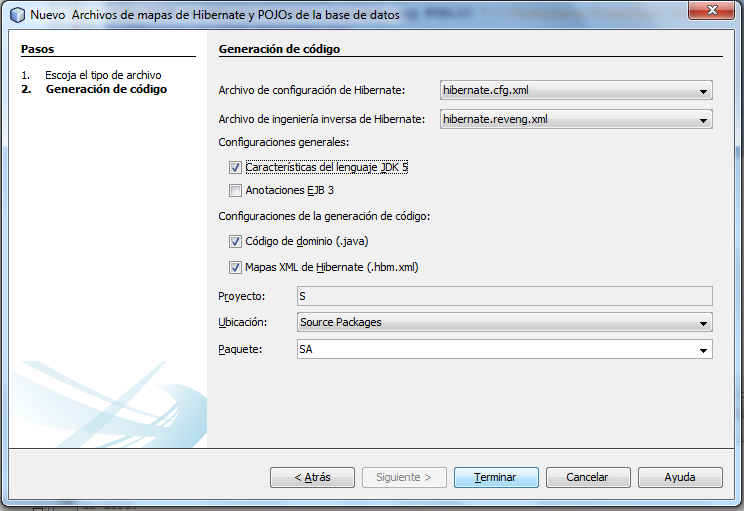
</hibernate-reverse-engineering>

Archivo de mapas de Hibernate y POJOs de la base de datos

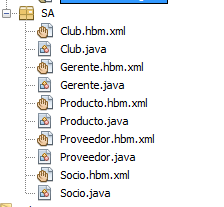
* Crea un Archivo nuevo.... Categoría Hibernar, tipo Archivos de mapas de Hibernate y POJOs de la base de datos.



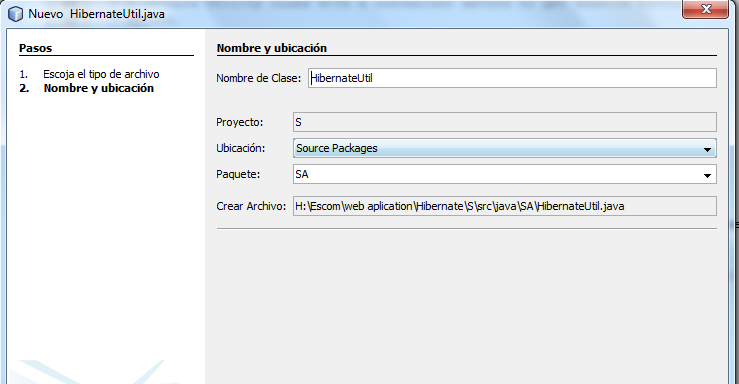
* Debes ponerle nombre a la clase donde estarán los mapas de Hibernate. Para que este tutorial funcione, ponle SA



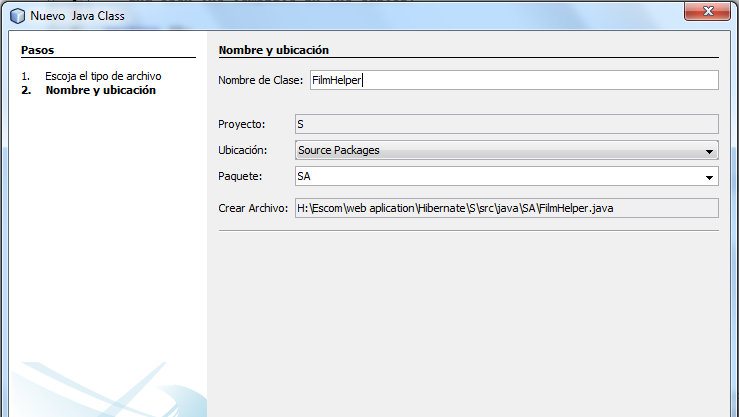
* Presiona el botón Terminar. Se deberían generar los siguientes archivos:



* Lo último sería crear un Archivo Nuevo..., Categoría Hibernar, Tipo HibernateUtil.java



* Crea una nueva Clase Java (Archivo Nuevo..., Categoría Java, tipo Clase Java) y llámale FilmHelper y ponlo dentro del paquete SA



* Modifica el código de FilmHelper de tal forma que quede como a continuación:

package SA;

**import org.hibernate.Session;**

public class FilmHelper {

Session session = null;

public FilmHelper() {

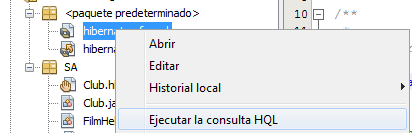
this.session = HibernateUtil.getSessionFactory().getCurrentSession();

}

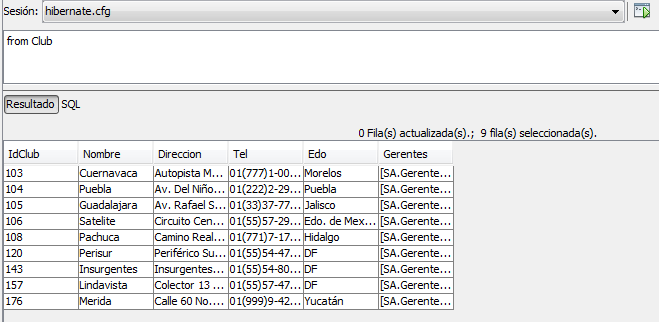
}

Consultas SQL

* Probemos algo. Haz click con el botón secundario sobre el archivo Source Packages/<paquete predeterminado>/hibernate.cfg.xml y selecciona la opción Ejecutar la consulta HQL del menú contextual



* En la barra de herramientas del editor de consultas HQL, selecciona la Sesión: hibernate.cfg y escribe lo siguiente en la primera caja de texto grande:



* Y presiona el botón que está al lado de la lista desplegable que debería estar mostrando hibernate.cfg como opción seleccionada para hacer la consulta y obtener un resultado como el siguiente